



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

DEPARTMENT OF COMPUTER SYSTEMS

IOT ŘEŠENÍ NA BÁZI SOC MT7688

IOT BASED ON SOC MT7688

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

JAKUB LUŽNÝ

VEDOUcí PRÁCE

SUPERVISOR

doc. Ing. ZDENĚK VAŠÍČEK, Ph.D.

BRNO 2018

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačových systémů

Akademický rok 2017/2018

Zadání bakalářské práce

Řešitel: **Lužný Jakub**

Obor: Informační technologie

Téma: **IoT řešení na bázi SoC MT7688**

IoT Based on SoC MT7688

Kategorie: Vestavěné systémy

Pokyny:

1. Seznamte se s platformou Mediatek MT7688AN, která nabízí IoT řešení ve formě systému na čipu (SoC). Zaměřte se na vlastnosti platformy, existující moduly integrující MT7688AN a možnosti programování.
2. Navrhněte vestavěný systém využívající SoC MT7688 poskytující Ethernet a WiFi rozhraní, který bude rozšířen o vhodný mikrokontrolér (např. STM32F103 v kombinaci s transceiverem CC1101), který bude zajišťovat bezdrátovou komunikaci v pásmu 868MHz.
3. Zpracujte studii na výše uvedené téma.
4. Navržený systém implementujte formou prototypu (např. formou modulu na DIN lištu) s možností napájení pomocí pasivního PoE standardu.
5. Vytvořte aplikaci demonstrující možnosti platformy (např. aktualizaci firmware mikrokontroléru přes webové rozhraní, příjem RF paketů).
6. Vyhodnoťte a diskutujte parametry navrženého řešení.

Literatura:

- Dle pokynů vedoucího.

Pro udělení zápočtu za první semestr je požadováno:

- Splnění bodů 1 a 2 zadání.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Vašíček Zdeněk, doc. Ing., Ph.D., UPSY FIT VUT**

Datum zadání: 1. listopadu 2017

Datum odevzdání: 16. května 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačových systémů a sítí
602 00 Brno, Božetěchova 2



prof. Ing. Lukáš Sekanina, Ph.D.
vedoucí ústavu

Abstrakt

Tato práce se zabývá vývojem vestavěného zařízení pro Internet věcí na platformě MediaTek MT7688A. Prozkoumává možnosti této platformy a dostupné řešení typu System on Module. Na základě těchto poznatků je implementován plně funkční prototyp takového zařízení. Pro předvedení možností zařízení je naprogramována webová aplikace, která zobrazuje data přijímaná na sběrnici Wireless M-BUS.

Abstract

This thesis deals with the development of an embedded device for Internet of Things based on the MediaTek MT7688A platform. It explores the capabilities of this platform and available System on Module solutions. Based on this findings, a fully functional prototype of such a device is implemented. To demonstrate abilities of the device, a web application is programmed that shows the data received over Wireless M-Bus bus.

Klíčová slova

Internet věcí, vestavěný Linux, OpenWRT, MediaTek, MT7688, Python, WMBUS, mikrokontrolér

Keywords

Internet of Things, Embedded Linux, OpenWRT, MediaTek, MT7688, Python, WMBUS, Microcontroller.

Citace

LUŽNÝ, Jakub. *IoT řešení na bázi SoC MT7688*. Brno, 2018. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. Ing. Zdeněk Vašíček, Ph.D.

IoT řešení na bázi SoC MT7688

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Zdeňka Vašíčka, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Jakub Lužný
16. května 2018

Poděkování

Tímto bych chtěl poděkovat zejména vedoucímu práce, panu doc. Ing. Zdeňku Vašíčkovi, Ph.D. za jeho odborné vedení, cenné rady a zapůjčení vývojových modulů. Také bych chtěl poděkovat mé rodině a přátelům za podporu při tvorbě této práce.

Obsah

1	Úvod	3
2	Platforma MT7688AN	4
2.1	Historie a podobné čipy	4
2.2	Moduly MT7688	5
2.2.1	LinkIt Smart 7688	5
2.2.2	AcSip AI7688H	6
2.2.3	Hi-Link HLK-7688A	6
2.2.4	Onion Omega2	7
2.3	Softwarová podpora	7
2.3.1	OpenWrt	8
2.3.2	U-Boot	8
3	Návrh systému	10
3.1	System on Module	11
3.2	Mikrokontrolér STM32F103C8T6	11
3.3	RF transceiver CC1101	12
3.4	Napájení	13
3.5	Ethernet	14
4	Implementace prototypu	15
4.1	Krabička a kompletace	15
4.2	Antény	16
4.3	Víko	17
4.4	Osazování	17
4.5	Cena	18
4.6	Zjištěné nedostatky	18
5	Firmware	19
5.1	OpenWRT	19
5.2	Problém s Flash pamětí	19
5.3	Ovladač WiFi	20
5.4	MCU	21
5.5	Demonstrační aplikace	21
5.5.1	Výběr programovacího jazyka	22
5.5.2	Implementace	22
5.5.3	Komunikace s MCU	23
5.6	Nahrání firmwaru do MT7688	23

5.6.1	Z běžícího systému	23
5.6.2	S pomocí bootloaderu	24
5.6.3	S pomocí Raspberry Pi	24
5.7	Nahrání firmwaru do MCU	25
6	Závěr	26
	Literatura	27
A	Obsah přiloženého paměťového média	29
B	Screenshot demonstrační aplikace	30
C	Fotografie výsledného prototypu	31
D	Seznam použitých součástek	32
E	Schéma zapojení	34

Kapitola 1

Úvod

Internet věcí (IoT) je termín pro moderní síť zařízení sahající od nejrůznějších čidel, přes nositelnou elektroniku až po domácí spotřebiče nebo průmyslovou automatizaci. Společným znakem těchto zařízení je schopnost komunikace s okolním světem, nejčastěji pomocí rádiových vln.

Pro získání a zpracování dat z těchto zařízení je třeba mít nějakou komunikační bránu, která umí přijímat odeslané pakety ze vzduchu a přeměrovat je dále ke zpracování pomocí standardní infrastruktury.

Na trhu existuje mnoho různých bran, které jsou obvykle uzavřené pouze pro komunikaci se zařízeními daného výrobce. Cílem této práce je vytvořit prototyp univerzální brány, která bude schopna komunikovat se zařízeními různých výrobců využívající různé protokoly. Díky tomu nebude muset uživatel vlastnit pro každého výrobce vlastní bránu, což povede k ušetření nákladů na pořízení jednotlivých bran, úspoře energie a jednodušší správě.

Tato brána by měla mít malé rozměry a možnost uchycení na DIN lištu, která se běžně vyskytuje v domovních rozvaděčích. Pro zapojení do existující infrastruktury by měla disponovat bezdrátovým rozhraním WiFi a také kabelovým rozhraním Ethernet. Brána bude koncipována tak, aby měla nízkou výrobní cenu a také nízkou spotřebu elektrické energie, což umožní její všestranné použití.

V první kapitole bude popsána platforma MT7688A a ekosystém kolem ní, druhá kapitola popisuje, jak probíhal návrh zařízení, které bylo ve třetí kapitole implementované. Poslední, čtvrtá kapitola, se věnuje vytvořenému firmwaru a demonstrační aplikaci, a způsobu jejich nahrání do zařízení.

Kapitola 2

Platforma MT7688AN

Čip MT7688AN od společnosti MediaTek je řešení typu SoC (System on Chip) integrující výkonný aplikační procesor společně s WiFi rádiem, ethernetovým přepínačem a dalšími periferiemi. Primárním zaměřením je trh s levnými domácími WiFi routery, ale díky velkému množství periférií je velmi vhodný i právě pro různé IoT aplikace. Blokové schéma čipu je na obrázku 2.1.

Základní parametry jsou:

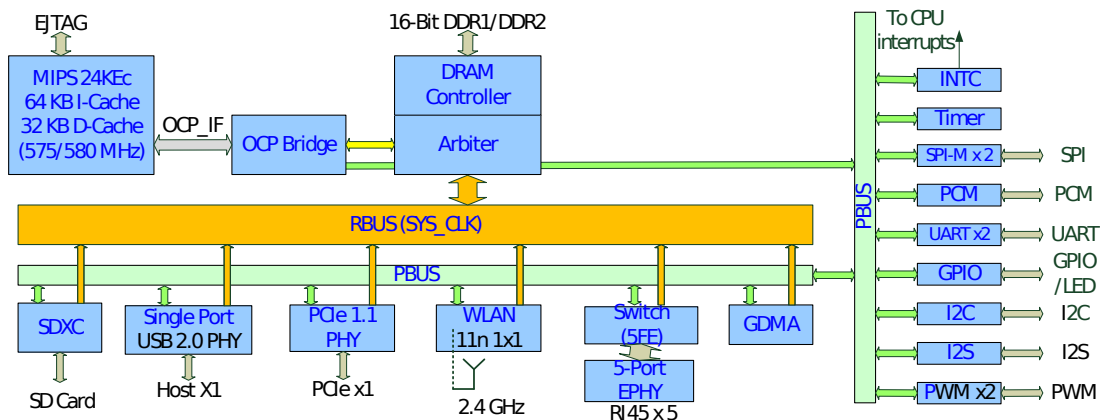
- 32bitový MIPS24KEc 580 MHz CPU
- podpora pro připojení až 256MiB DDR nebo DDR2 RAM
- 1T1R 802.11n 2,4GHz rádio (150Mbps)
- integrovaný 5portový Fast Ethernet switch
- 1 linka rozhraní PCIe v1.1, 1 port USB 2.0 Host, 1x eMMC/SDXC
- rozhraní UART 3×, SPI Master, SPI Slave, I²C, I²S
- 4 kanály PWM generátoru
- podpora operačního systému Linux, eCos a VxWorks

Zvláštností tohoto procesoru je, že umí pracovat ve dvou tzv. módech, které se liší tím, které periferie procesoru jsou dostupné. Zatímco v módu *IoT Device* má procesor vyveden pouze jeden ethernetový port, v módu *IoT gateway* jich má vyvedeno pět. Přejde tím ale o některé periferie, které jsou vyvedeny na stejných pinech jako ethernetové páry. Konkrétně tím přijde o rozhraní eMMC/SD, SPI slave, port UART2 a také dva PWM kanály.[9]

2.1 Historie a podobné čipy

Tchaj-wanská společnost MediaTek se na trhu s řešeními pro WiFi routery objevila v roce 2011 díky akvizici společnosti Ralink. Pod křídly MediaTeku se řada RT2xxx/RT3xxx vyvinula do řady MT76xx. Předchůdcem čipu MT7688 byl MT7620, který měl méně periférií a menší energetickou účinnost.

MT7688 se vyrábí ve dvou provedeních - kromě MT7688A existuje také verze s integrovanou 8MiB pamětí RAM a bez možnosti připojení SD karty, označovaná jako MT7688K. Dále se také vyrábí čip MT7628, který se od MT7688 liší použitím rychlejšího 2T2R rádia (300Mbps).[15]



Obrázek 2.1: Schéma procesoru MT7688A v režimu *IoT Gateway*. Převzato z datasheetu[9].

2.2 Moduly MT7688

Pro prototypové nebo malosériové aplikace se typicky v zařízeních nepoužívá přímo samotný procesorový čip, ale takzvaný System on Module (SoM). Jedná se o desku plošného spoje, která na sobě kromě samotného procesoru integruje další součástky nezbytné k jeho běhu, jako je operační paměť a obvody řídicí napájení procesoru, a také různé pasivní komponenty (blokovací kondenzátory, referenční rezistory apod.). Někdy bývá obsažena i Flash paměť pro uložení operačního systému.

Použitím takového modulu si vývojář může ušetřit starosti s nákupem velkého množství různých součástek a vyhne se osazování procesoru a operační paměti, které jsou typicky v BGA pouzdrech a tudíž je lze osadit jen pomocí speciálního vybavení. Díky tomu, že výrobce modulu nakupuje součástky v obrovských sériích, je pak často cena modulu podobná, jako by byla celková cena jednotlivých obsažených komponent. Nevýhodou bývá omezená nabídka periférií procesoru, protože zpravidla nejsou vyvedeny veškeré signály procesoru, případně jsou některé už použité přímo na modulu.

Nabídka modulů pro platformu MT7688 je poměrně široká, ale většina z nich je, stejně jako procesor, asijského původu a často tak naráží na problém s dostupností, kvalitou a hlavně dokumentací. V této kapitole budou rozebrány jednotlivé moduly z hlediska hardwarové výbavy, vyvedených signálů, rozměrů, dostupnosti a kvality dokumentace. Uvedeny budou také orientační ceny v amerických dolarech. Srovnání základních parametrů modulů je v tabulce 2.1.

2.2.1 LinkIt Smart 7688

Jedná se o oficiální vývojovou platformu vyvinutou v rámci spolupráce mezi společnostmi MediaTek a Seeed Studio. Z toho plyne poměrně kvalitní dokumentace, zahrnující kompletní schéma zapojení a návrh PCB ve formátu PDF. Platforma má také vlastní diskuzní fórum, které ovšem staví spíše na komunitní spolupráci než oficiální podpoře. Dostupnost je velmi dobrá, lze objednat přímo přes Seeed Studio a je také dostupný v běžných distribučních kanálech.

Modul disponuje 128MiB DDR paměti a 32MiB paměti Flash. Přímo na modulu je také slot na microSD kartu a dva microUSB konektory - jeden slouží pro přivedení napájení, druhý jako USB hostitel. Jeho rozměry jsou $56\text{mm} \times 26\text{mm} \times 18\text{mm}$. Vyrábí se také ve variantě Duo, která přidává mikrokontrolér Microchip (dříve Atmel) ATmega32U4 s podporou

Arduino. Existuje k němu Breakout Board, do kterého se modul zapojí a je z něj vyveden jeden Ethernetový port, USB hostitel jako konektor typu A, dále je přítomen zvukový kodek, který převádí I²S výstup procesoru na analogový signál vyvedený do standardního 3,5mm konektoru Jack.

Procesor je standardně nastaven bootloaderem do režimu *IoT device* a má vyvedeny rozhraní 1× Ethernet, 3× UART, USB, SPI Master i Slave, I²S a I²C. Po přepnutí do režimu *IoT gateway* by měl 3× Ethernet, 2× UART, USB, SPI Master, I²S a I²C. Kromě toho by také přestal fungovat slot na microSD kartu.

Nevýhodou tohoto modulu je jeho velikost a vyvedení signálů pomocí pinových lišt, což dále zvyšuje jeho prostorové nároky v případě osazení na vlastní PCB. Proti hraje také cena - cca \$15.[16]

2.2.2 AcSip AI7688H

Jde o modul tchaj-wanského výrobce AcSip, který se specializuje na řešení typu SoM v oblasti IoT[11]. Modul lze zakoupit na webu Seeed Studio, z běžných distributorů jej mají např. DigiKey nebo Mouser. Co se dokumentace týče, je k dispozici pouze oficiální Product Brief, který popisuje základní parametry a seznam vyvedených signálů.

Na modulu staví projekt bezdrátového reproduktoru ReSpeaker Core. Ten je open-source včetně návrhu HW z programu Eagle, tudíž by se z něj dalo čerpat např. rozmístění plošek pro zapájení modulu, které není v dokumentaci nijak zakreslené.[7]

Výrobce má svou stránku na GitHubu, kde jsou k dispozici např. zdrojové kódy bootloaderu a OpenWRT. Vyvedeny jsou následující signály: 5× Ethernet, 2× UART I²S, I²C, USB a PCI Express.

Na modulu je osazeno 128MiB paměti a 32MiB paměti Flash. Kromě toho na modulu nalezneme pouze u.FL konektor pro připojení WiFi antény. Veškeré signály jsou vyvedeny pomocí pájecích plošek na bočních stranách. Jako vývojovou platformu nabízí výrobce EK-AI7688H. Jedná se o desku, která z modulu vyvádí jeden Ethernetový port, USB a SMA konektor pro anténu. Kromě toho jsou také vyvedeny signály z pájecích plošek na pozice pro pinovou lištu. Na webu AcSip je tato deska nabízena za \$150 a jinde není běžně dostupná.

Cena modulu je opět cca \$15. Jeho rozměry jsou 32mm × 24mm × 2,5mm.

2.2.3 Hi-Link HLK-7688A

Svůj modul vyrábí také čínská firma Hi-Link[13]. Lze je zakoupit na čínském e-shopu AliExpress za cenu okolo \$10. Na webu výrobce jsou zdokumentovány základní parametry, seznam vyvedených signálů a dá se nalézt i nákres rozmístění jednotlivých plošek. Po zažádání výrobce nám byla zaslána rozsáhlejší dokumentace, zahrnující schéma zapojení modulu s vynechanými hodnotami součástek.

Jsou vyvedeny tyto signály: 5× Ethernet, 2× UART, I²S, I²C, USB a SPI. Rozhraní SPI je vyvedeno včetně signálu SPI_CS0, který slouží jako chip select pro interní Flash paměť. Díky tomu je možné při držení procesoru v resetu komunikovat s Flash pamětí z vnějšku. To se může hodit pro nahrání firmware např. při poškození bootloaderu.

Modul je vybaven 128MiB paměti a 32MiB paměti Flash. Kromě plechu kryjícího veškeré součástky je na horní straně modulu pouze u.FL konektor pro připojení antény a zelená LED signalizující aktivitu WiFi. Rozměry činí 32,8mm × 18mm × 2,8mm.

Výrobce nabízí vývojovou platformu nazvanou HLK-7688A Startkit, která z modulu vyvádí 5 ethernetových portů a 2 sériové porty. Ta je dostupná na AliExpressu za cca \$20.

	LinkIt Smart	AI7688H	HLK-7688A	Omega2+	Omega 2S+
Rozměry [mm]	56 × 26	32 × 24	32,8 × 18	42,9 × 26,4	34 × 20
Výška [mm]	18	2,5	2,8	9,9	2,8
Cena	\$15	\$15	\$10	\$9	neznámá
Kontakty	pinové lišty	SMD plošky	SMD plošky	pinové lišty	SMD plošky
RAM	128MiB	128MiB	128MiB	128MiB	128MiB
Flash	32MiB	32MiB	32MiB	32MiB	32MiB
Ethernet	3×	5×	5×	1×	5×
UART	2×	2×	2×	2×	2×
PCI Express	ne	ano	ne	ne	ano
SPI Master	ano	ano	ano	ano	ano
I ² S	ano	ano	ano	ano	ano
I ² C	ano	ano	ano	ano	ano

Tabulka 2.1: Srovnání jednotlivých modulů. V případě více variant byla vybrána ta vybavenější. Počty vyvedených rozhraní jsou vždy uvedeny v módu procesoru *IoT gateway*.

2.2.4 Onion Omega2

Jedná se o projekt, který začal jako crowdfundingová kampaň na serveru Kickstarter s heslem „\$5 Linux Computer with Wi-Fi, Made for IoT“. Jedná se o nástupce předchozího projektu Onion Omega poháněného čipem Qualcomm Atheros AR9331, který měl ovšem cenovku \$20.

Omega2 je vybavena 64MiB paměti RAM a 16MiB paměti Flash.[20] Prodává se také varianta Omega2 Plus, která je vybavena 128MiB RAM a 32MiB Flash a navíc přidává MicroSD slot na spodní straně modulu. Základní Omega2 se aktuálně prodává za \$7,5, verze Plus za \$9. Oba jsou dostupné v běžných distribučních kanálech, kromě toho i v e-shopech zaměřených na amatérskou komunitu.

Modul má signály vyvedené na pinových lištách. Je koncipován jako modulární platforma, existuje pro něj několik různých základních desek. Do některých se dají navíc přidávat rozšiřující moduly. Vyvedeny jsou rozhraní Ethernet, USB, 2× UART, SPI, I²S a I²C. Tato kombinace neumožňuje ani po přepnutí procesoru do režimu *IoT gateway* vyvedení více než jednoho ethernetového portu. Rozměry jsou 42,9mm × 26,4mm × 9,9mm.

Dokumentace obsahuje základní parametry a seznam vyvedených signálů, je také dostupná rozsáhlá uživatelská dokumentace. Schéma zapojení není k dispozici.

Zajímavou je varianta Omega2S nebo Omega2S+.[19] Hardwarové parametry jsou stejné jako u běžných Omega2, ale pinové lišty jsou nahrazeny pájecími ploškami a vyvedena jsou všechna rozhraní procesoru. Zajímavostí je samostatné vyvedení napájení pro Flash paměť, což by se teoreticky dalo využít k připojení vlastní paměti. Tato varianta ovšem není běžně dostupná, na webu výrobce je k dispozici pouze vývojový kit obsahující 4 moduly, základní desku a příslušenství za \$249. Rozměry této varianty modulu jsou 34mm × 20mm × 2,8mm

2.3 Softwarová podpora

Přestože platforma MT7688 by se dala, stejně jako jakákoliv jiná platforma, programovat zcela bez operačního systému (tzv. bare metal, podobně jako mikrokontroléry), jeho použití umožňuje určitou míru abstrakce. Síla MT7688 spočívá zejména v podpoře otevřeného operačního systému Linux. Přímou od výrobce je v rámci programu LinkIt Smart podporo-

vaná linuxová distribuce OpenWRT. K zavedení linuxového jádra je od výrobce podporován bootloader U-Boot.

2.3.1 OpenWrt

OpenWrt je snadno rozšiřitelná distribuce Linuxu pro vestavěná zařízení (zejména bezdrátové směrovače). Na rozdíl od jiných distribucí pro tato zařízení byla od začátku postavena tak, aby byla plně funkční a snadno upravitelný operační systém pro domácí směrovače.

Místo toho, aby se snažilo vytvořit jeden statický firmware, poskytuje OpenWrt plně zapisovatelný souborový systém s volitelným balíčkovacím manažerem. To osvobozuje uživatele od výběru aplikací a konfigurace dodané výrobcem a umožňuje mu přizpůsobit si pomocí balíčků vestavěné zařízení tak, aby vyhovovalo jakékoliv aplikaci. Pro vývojáře poskytuje OpenWRT framework, pomocí kterého lze vytvořit balíček bez nutnosti sestavení celého firmware.

V roce 2016 vznikl projekt LEDE (Linux Embedded Development Environment) jako odnož OpenWRT. Kladl si za cíl vytvořit pravidelné cykly vydávání nových verzí, transparentní rozhodovací proces a širší zapojení komunity. K projektu se připojila významná část neaktivnějších vývojářů OpenWRT. V lednu 2018 se oba projekty dohodly pro opětovné spojení pod názvem OpenWrt. Spojený projekt vychází z kódu LEDE a přejímá jeho pravidla, procesy a infrastrukturu.[2]

Pro konfiguraci se používá centralizovaný systém UCI (Unified Configuration Interface). Všechny konfigurační soubory jsou umístěny v adresáři */etc/config/* a jedná se o textové soubory s jednotným formátem. Používají se jak ke konfiguraci nástrojů spadajících přímo pod OpenWRT (např. konfigurace síťových rozhraní), tak ke konfiguraci doinstalovaných aplikací (např. OpenVPN), kdy se před spuštěním aplikace převede formát UCI na nativní konfigurační soubor pro danou aplikaci. Konfigurace systému lze provádět přímou editací těchto souborů, případně konzolovým nástrojem *uci*, který se hodí zejména pro použití ve skriptech. Na tomto systému také staví webové konfigurační rozhraní LuCI, které je naprogramováno v jazyce Lua a poskytuje uživatelsky přívětivé rozhraní pro nastavení zařízení. Kromě toho v něm lze zjistit také různé stavové informace nebo zobrazit graf datového toku na síťovém rozhraní.

2.3.2 U-Boot

Das U-Boot je zavaděč systému určený pro vestavěné systémy postavené na procesorech PowerPC, ARM, MIPS a dalších.[8] Jedná se o svobodný software naprogramovaný v jazyce C a dostupný pod licencí GNU GPL. Může být instalován do bootovací ROM paměti (typicky však paměti Flash) a slouží k inicializaci a otestování hardwaru a také k stažení a spuštění aplikačního kódu.

Vývoj U-Bootu probíhá v těsné souvislosti s Linuxem: některé části zdrojového kódu z Linuxu přímo pochází a jsou také sdíleny některé hlavičkové soubory. Speciální důraz je kladen na podporu zavádění obrazů linuxového jádra. Software se však snaží být snadno konfigurovatelný a rozšiřitelný.

Je schopný zavést operační systém přímo ze souborového systému (podporuje většinu souborových systémů linuxového světa), nebo také bez souborového systému při zadání pozice a velikosti obrazu v paměti. Je schopný číst i zapisovat paměti typu Flash připojené přímo k procesoru, ale také USB disky a zvládne i získat operační systém přímo ze sítě protokolem TFTP.

Jeho konfigurace se ukládá to tzv. prostředí (environment). Jedná se sadu proměnných, uložených textově za sebou jako název a hodnota, oddělených rovnítkem. Každá proměnná je fakticky program v jednoduchém skriptovacím jazyce, který může využívat vestavěných příkazů a hodnot ostatních proměnných.

Kapitola 3

Návrh systému

Pro splnění požadavků na systém bylo nutné vybrat komponenty, které umožní realizovat jeho funkce a navrhnout jejich propojení.

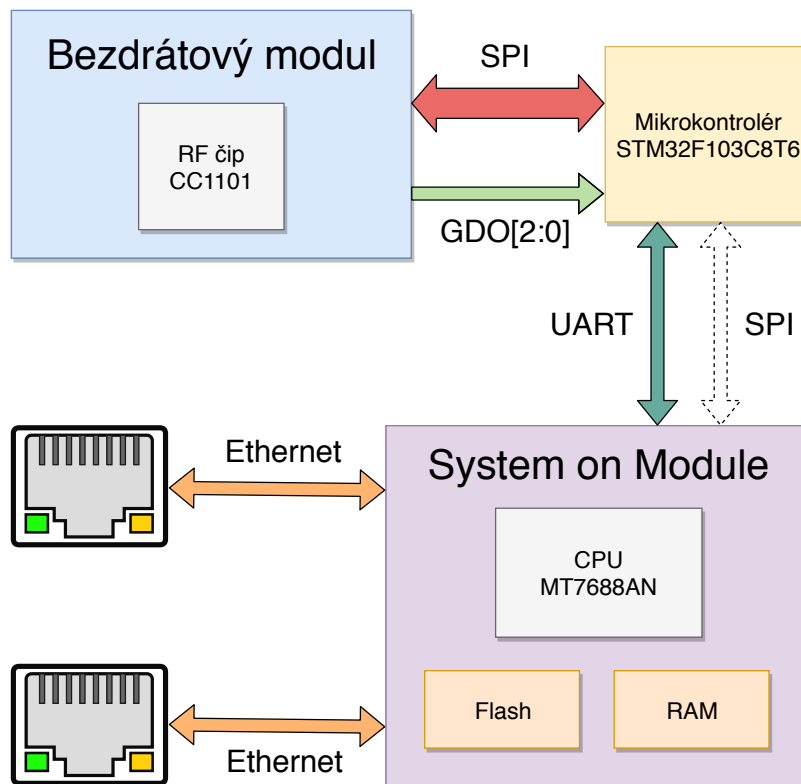
Použití čipu MediaTek MT7688A jako hlavního procesoru, na kterém poběží operační systém a aplikační logika, vyplývá už ze zadání. Vzhledem k použití operačního systému však není vhodné tento procesor používat k operacím, které je nutné vykonávat v reálném čase, jako je reakce na přerušení z RF transceiveru. Z toho důvodu bylo rozhodnuto o použití mikrokontroléru, který bude zprostředkovávat komunikaci mezi RF transceiverem a procesorem.

Pro tento účel byl vybrán mikrokontrolér STM32F103C8T6. Důvodem pro výběr tohoto čipu byla jeho snadná dostupnost a také možnost prototypování na levných vývojových deskách, zejména takzvané „Blue pill.“[3]. Disponuje také dostatečným výpočetním výkonem a pamětí, množstvím komunikačních rozhraní a jeho pouzdro se dá snadno ručně osadit.

Komunikace na pásmu ISM 868MHz vyžaduje použít čip, který zajistí převod digitálních dat na analogový signál a jeho zesílení v případě vysílání, a opačný proces v případě příjmu. Byl vybrán čip CC1101 od společnosti Texas Instruments, který je velmi často používán pro tyto účely, např. v různých open-source projektech zaměřených na Smart Home. Je k němu také dostupná kvalitní dokumentace a software SmartRF® Studio, který umožňuje vygenerovat optimální nastavení registrů čipu pro zadané parametry přenosu.

Rozhraní čipu RF transceiveru, pomocí kterého se připojuje k řídicímu čipu (v našem případě mikrokontroléru), je SPI. Paralelně budou vyvedeny také signály GDO0-GDO2, které slouží jako signály přerušení a jejich význam se dá konfigurovat. Pro spojení mikrokontroléru s hlavním procesorem bude použito rozhraní UART, zejména kvůli možnosti oboustranné inicializace komunikace. Mikrokontrolér má v sobě navíc bootloader, který umožňuje nahrání programu pomocí rozhraní UART. Pro možnost dosažení vyšší rychlosti bude alternativně vedeno i rozhraní SPI společně s jedním signálem přerušení, kterým by mohl mikrokontrolér indikovat potřebu zahájení komunikace.

Rozhraní Ethernet je z procesoru vyvedeno už přímo ve formě tzv. MDI (Medium Dependent Interface), tudíž nepotřebuje externí řadič fyzické vrstvy. Metalické rozhraní ovšem pro svou funkčnost požaduje galvanické oddělení, které se běžně realizuje transformátory. Lze také použít konektor RJ45, který má transformátory integrované. Celkové zapojení je vyjádřeno na blokovém schématu (obrázek 3.1). V následujících podkapitolách budou detailněji popsány jednotlivé komponenty a jejich zapojení. Schéma zapojení prototypu je přiloženo v příloze E.



Obrázek 3.1: Blokové schéma navrženého řešení

3.1 System on Module

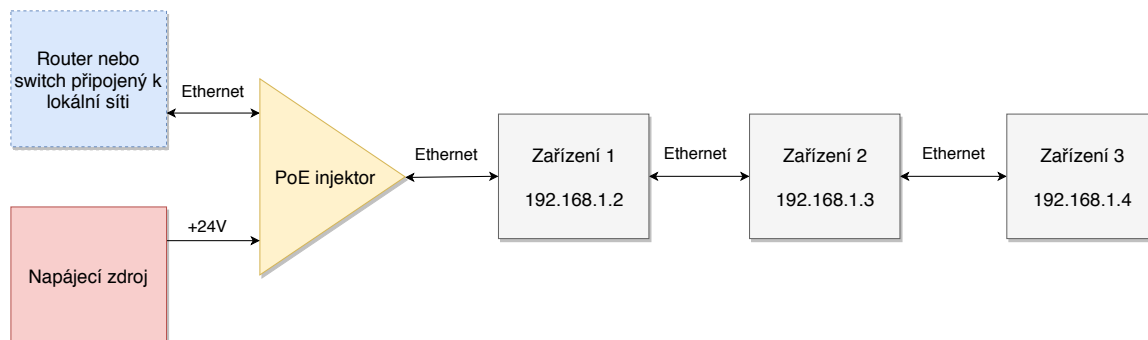
Jako systémový modul byl po dlouhém rozhodování zvolen Hi-Link HLK7688A. Důvodem pro tuto volbu bylo zejména vyvedení signálů pomocí pájecích plošek, ale také jeho kompaktní rozměry, cena a dostupnost. Při návrhu zapojení se vycházelo ze schématu k platformě LinkIt Smart 7688 a také ze schématu samotného modulu dodaného jeho výrobcem.

3.2 Mikrokontrolér STM32F103C8T6

Mikrokontrolér STM32F103C8T6 patří do rodiny STM32 výrobce STMicroelectronics. Tato rodina zakládá na 32bitových výpočetních jádrech architektury ARM Cortex-M, které jsou spojeny s operační pamětí SRAM, pamětí Flash pro kód a různými periferiemi v jednom pouzdře.^[23]

Základní parametry čipu STM32F103C8T6 jsou:

- 32bitové jádro ARM Cortex-M3 s frekvencí až 72MHz
- 20KiB paměti SRAM, 64KiB paměti Flash
- 4 časovače, 37 pinů pro GPIO
- dva 12-bitové AD převodníky schopné obsluhovat 10 kanálů
- komunikační rozhraní SPI 2×, I²C 2×, USART 3×, USB a CAN



Obrázek 3.2: Schéma řetězení zařízení pomocí integrovaného ethernetového switchu a průchozího PoE napájení

Návrh zapojení mikrokontroléru vycházel zejména z oficiální návrhové příručky[22] a datasheetu.[23] Bylo třeba dát si pozor na správné nastavení pinů BOOT0 a BOOT1 pro aktivaci bootloaderu. Pro jeho funkčnost je také potřeba použít port UART1 a to konkrétně na pinech PA9 a PA10. K mikrokontroléru jsou také připojeny tři LED diody indikující jeho aktivitu. Zelená dioda indikuje příjem paketu, oranžová jeho odeslání a červená je stavová. Pro ladění při vývoji bylo také vyvedeno rozhraní SWD (Serial Wire Debug) na pinovou lištu společně s jedním portem UART.

Jak již bylo zmíněno dříve, jako alternativa k rozhraní UART je mikrokontrolér připojen k procesoru taky rozhraním SPI společně s jedním signálem přerušení. Všechna tato propojení byla realizována tak, že cestičky jsou přerušené a pro jejich aktivaci je nutné osadit 0R rezistor.

3.3 RF transceiver CC1101

Čip CC1101 od společnosti Texas Instruments je nízkopříkonový RF transceiver. Slouží ke komunikaci v pásmech ISM¹, což jsou bezlicenční pásma, tzn. za dodržení určitých podmínek provozu je možné v nich vysílat bez nutnosti získání licence nebo oznámení úřadům. Toho se často využívá pro bezdrátové senzorové sítě, zabezpečovací zařízení a také domácí a průmyslovou automatizaci. Příkladem může být sběrnice Wireless M-BUS.

Parametry transceiveru jsou[24]:

- komunikační frekvence 315/433/868/915MHz
- datová rychlost 0,6 až 600 kilobit/s
- podpora modulací 2-FSK, 4-FSK, GFSK, MSK a OOK
- komunikační rozhraní SPI

Pro zjednodušení návrhu a dosažení lepších parametrů nebyl použit přímo čip CC1101, ale byl zvolen rádiový modul Anaren A1101R08C[12]. Tento konkrétní modul byl zvolen zejména z důvodu kompaktních rozměrů. Má na sobě konektor u.FL pro externí anténu, ale vyrábí se také verze A1101R08A s anténou integrovanou na modulu. Pro použití integrované antény by však bylo nutné pozměnit návrh plošného spoje spočívající v odstranění veškeré

¹Industrial, Scientific and Medical

mědi pod modulem. Výhodou tohoto modulu je také možnost nahrazení jiným typem, který bude naladěný na jinou frekvenci.

3.4 Napájení

Návrh napájení byl zkomplikován skutečností, že výrobce v datasheetu MT7688 neudává jeho spotřebu. Bylo tedy nutné změřit spotřebu vývojové platformy LinKit Smart, která se pohybovala kolem 1W. To je příliš mnoho na to, aby se zařízení dalo napájet bateriemi rozumných rozměrů.

Bylo tedy nutné do zařízení dovést externí napájení. Vzhledem k přítomnosti rozhraní Ethernet se nabízelo jej pro napájení použít. To je možné díky konceptu Power over Ethernet, který je definován v normě IEEE 802.3af. Pro napájení se používají standardní ethernetové kabely a navíc není nijak narušena datová komunikace. Tato norma však využije použití čipu typu PD (Powered Device), který komunikuje s napájecím zdrojem (PSE - Power Sourcing Equipment). Díky tomu je možné detekovat, zda je zařízení připojené ke zdroji a teprve v tu chvíli mu začít dodávat proud. V případě, že se k takovému zdroji připojí zařízení, které PoE nepodporuje, proud nebude dodáván a nemůže tak dojít k poškození zařízení, které není na tento druh napájení připraveno.

Nevýhodou napájení pomocí standardu IEEE 802.3af (tzv. aktivní PoE) je však nutnost použití dalšího čipu na straně zařízení, což komplikuje návrh a zvyšuje cenu. Proto existuje alternativa označovaná jako pasivní PoE, která využívá dvou nevyužitých páru ethernetového kabelu při komunikační rychlosti 100Mbps. Tato koncepce je běžná u levných síťových zařízení, např. od výrobců Mikrotik a Ubiquity. Z důvodu kompatibility s těmito výrobci bylo zvoleno napájecí napětí 24V.

Díky jednoduchosti pasivního PoE se dá snadno implementovat také průchozí řešení, kdy napájecí napětí přivedené jedním konektorem RJ45 bude dále distribuováno do dalšího konektoru. To v kombinaci s integrovaným ethernetovým switchem v procesoru umožní řetězit tyto zařízení za sebou. Schéma takového řešení je ilustrováno na obrázku 3.2.

V zařízení tedy bylo nutné převést napětí 24V přivedené pomocí PoE na napětí 3,3V, které vyžadují veškeré komponenty. Z důvodu velkého rozdílu mezi vstupním a výstupním napětím byl použit spínaný zdroj. Pro tyto účely byl vybrán čip LMR16020 od společnosti Texas Instruments. Jedná se o měnič typu step-down, který vyniká velkým rozsahem vstupního napětí (až 65V), dostatečným maximálním výstupním proudem (2A) a nízkým počtem nutných podpůrných komponent. Při návrhu se vycházelo z kapitoly *Application and Implementation* datasheetu.[25]

Napětí přivedené do zařízení pomocí volných párů v Ethernetovém kabelu nejprve projde přes usměrňovací můstek. Tím je zajištěna ochrana proti přepólování v případě použití nestandardního zapojení kabelu. Následně je do cesty zařazen transil, který má za účel zlikvidovat napěťové špičky, které mohou vzniknout při připojení kabelu. Po vyfiltrování na vstupním kondenzátoru jde napětí do samotného zdroje. Ten je nastaven na spínací frekvenci cca 500KHz, což umožňuje použití menších komponent. Po vyfiltrování na cívce a kondenzátoru je získáno napětí 3,3V.

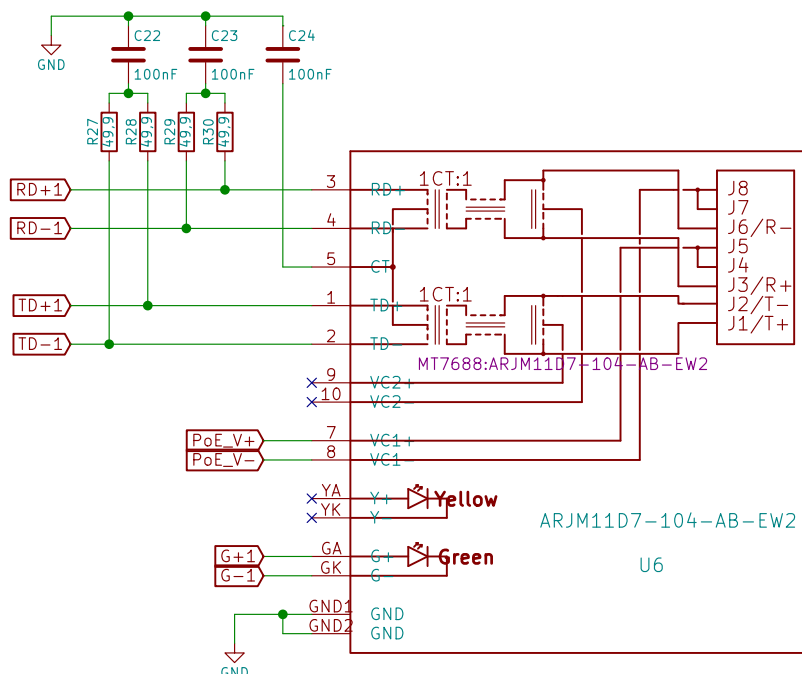
Průchodnost napájení přes PoE je řešena prostým propojením párů vyhrazených pro napájení mezi konektory. Napájení pomocí stejnosměrné složky v datových párech se v prototypu nepoužívá.

3.5 Ethernet

Z důvody úspory plochy bylo rozhodnuto o použití Ethernetových konektorů RJ45 s vestavěnými transformátory. Ty bylo nutné vybrat tak, aby měly podporu PoE, tzn. měly vyvedené páry 4,5 a 7,8 vyvedené napřímo, bez transformátorů v cestě. Některé PoE konektory ovšem mají v sobě integrovaný i usměrňovací můstek, což by zabránilo možnosti průchodu PoE, proto bylo třeba se jim vyhnout.

Většina konektorů s integrovanými transformátory však bývá větších rozměrů, zejména co se týče hloubky zástavby. Vzhledem k tomu, že šířka procesorové desky je daná prostorem, který vznikne mezi těmito konektory, bylo nutné nalézt konektor s co nejmenší hloubkou. Požadavkům vyhověl konektor ARJM11D7-104-AB-EW2 od společnosti Abracon s hloubkou pouze 21,3mm.[10].

Fyzická vrstva integrovaná v procesoru vyžaduje použití externích terminačních rezistorů na všech Ethernetových signálech. Střed transformátorů by měl být připojen na GND přes kondenzátor. Detail zapojení je na obrázku 3.3.



Obrázek 3.3: Schéma zapojení Ethernetu. Diferenciální signály TD a RD vedou přímo do procesoru.

Kapitola 4

Implementace prototypu

Tato kapitola se bude zabývat návrhem PCB a řešením mechanické montáže prototypu. Pro návrh elektroniky byla použita sada open-source aplikací KiCad ve verzi 4.0.7. Výsledné návrhy jsou přiloženy na médiu, které je součástí práce.

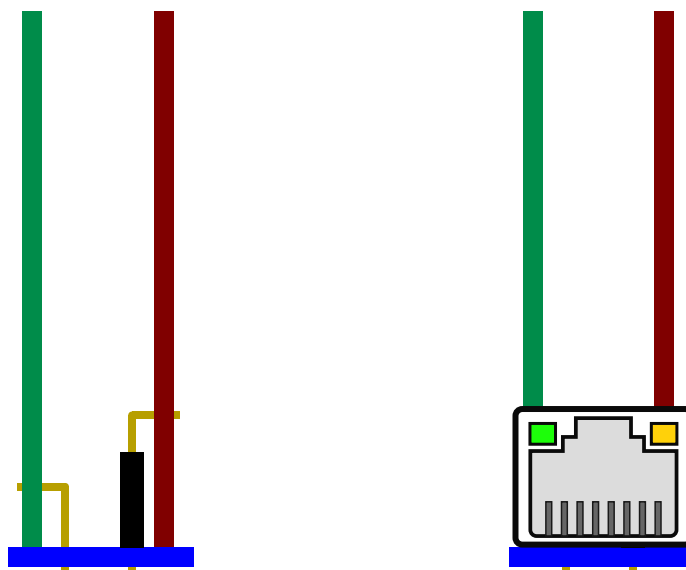
4.1 Krabíčka a kompletace

Výsledné zařízení by mělo jít namontovat na standardní DIN lištu, na kterou je k dispozici široký sortiment krabiček. Tyto krabičky se velikostně dělí do kategorií podle toho, kolik tzv. modulů na DIN liště zaberou (tzn. 1M, 2M, 3M atd.) Pro představu, ve velikosti 1M jsou běžně vyráběny domácí jističe. Zbylé dva rozměry už záleží na výrobcí a modelu krabičky, ale vzhledem k tomu, že by výsledný produkt měl jít zamontovat do běžného rozvaděče, by se měly udržet co nejmenší.

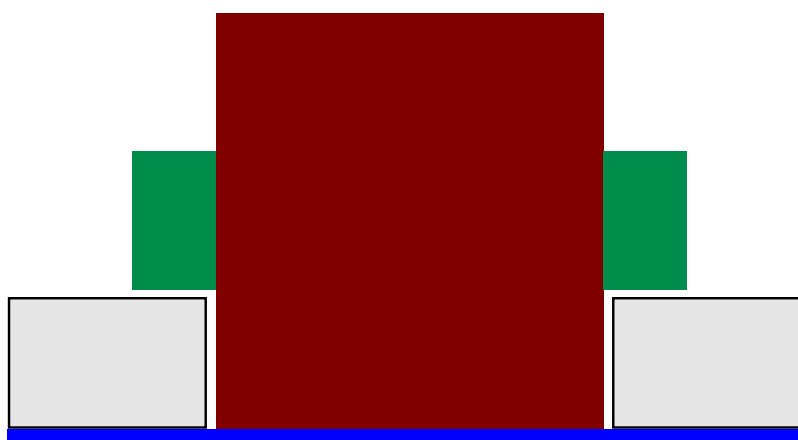
Vzhledem k velikosti součástek, které byly do zařízení navrhnuty, a to zejména modulu SoM, bylo třeba vybrat takovou krabičku, do které se celé zařízení rozumně umístí. Nelze taky opomenout nutnost rozumně vyvést dva RJ45 konektory. Z toho důvodu jsem se nejprve vydal cestou širší krabičky, do které by se všechny součástky a konektory naskládaly na jednu PCB umístěnou v rovině s DIN lištou. To by ovšem znamenalo velikost krabičky minimálně 3M. Proto jsem začal hledat alternativy.

Jako nejvhodnější se jevílo použít 1M krabičku 25.0104RJ0.BL řady Modulbox XTS italské firmy Italtronic[1]. Ta má od výroby vyfrézované dvě pozice pro RJ45 konektory a umožňuje umístění PCB v několika patrech, a to v obou rovinách. Napadlo mě tedy řešení, že v krabičce bude umístěna jedna PCB v rovině s DIN lištou, ze které budou vyvedeny konektory RJ45, a do ní budou kolmo zapadat dvě PCB. Zkusil jsem si tedy začít skládat součástky na jednotlivé PCB a dospěl k řešení, kdy na jedné PCB kolmé k rovině DIN lišty bude CPU modul, rádiový modul a mikrokontrolér včetně veškerých podpůrných součástek a na druhé kolmé PCB bude umístěn napájecí zdroj. Tyto desky se propojí přes třetí desku s konektory RJ45, umístěnou v rovině DIN lišty. Propojení jsem vyřešil pomocí pinových lišt pro možnost přizpůsobení vzájemných pozic PCB hloubkou zapájení těchto lišt. Na obou PCB s elektronikou budou pinové lišty, jako protikus budou v případě zdrojové PCB sloužit pinové dutinky. To umožní nahrazení zdrojové PCB jiným zdrojem, např. laboratorním, a taky snadný přístup ke všem součástkám pro případné měření. PCB s procesorem bude připájena napevno, jako protikus jí budou sloužit pouze vyvrtané díry. Řez touto sestavu z různých úhlu je na obrázcích 4.1 a 4.2. Pro lepší představu je na obrázku 4.4 fotografie

spojených PCB. Fotografie finálního zařízení v krabičce s připojenou anténou je v příloze C.



Obrázek 4.1: Nákres bočního pohledu na sestavené PCB. Vlevo jsou vynechány konektory RJ45 pro zvýraznění způsobu propojení. Zelená PCB je procesorová, modrá propojovací s konektory RJ45 a červená zdrojová.



Obrázek 4.2: Nákres čelního pohledu na sestavené PCB. Zelená PCB je procesorová, modrá propojovací s konektory RJ45 a červená zdrojová.

4.2 Antény

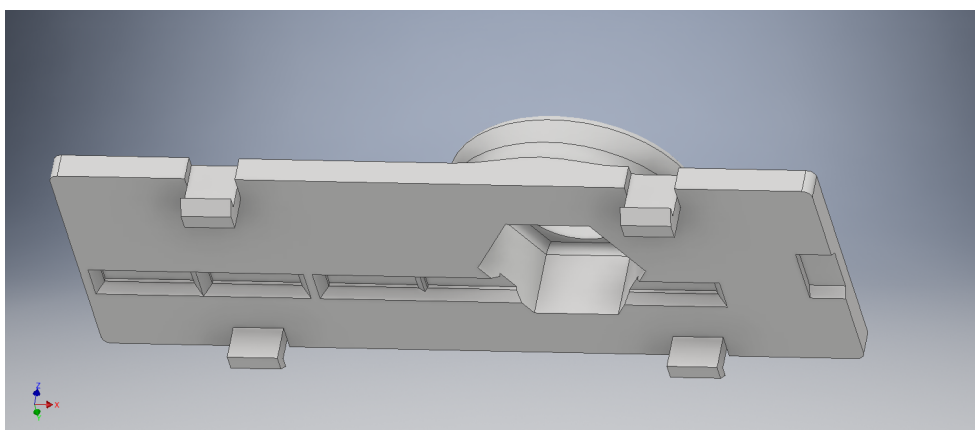
Zařízení obsahuje dvě součástky komunikující pomocí rádiových vln – kromě RF transceiveru CC1101 také rozhraní WiFi v procesoru MT7688. Obě tyto zařízení potřebují pro svou funkčnost anténu. U WiFi stačí díky jeho vyšší frekvenci (2,4GHz) anténa menší, byla proto použita plochá samolepící anténa, která se nalepí na vnitřní stranu PCB se zdrojem. Ta byla proto při návrhu v těchto místech zbavena mědi. Pro komunikaci v pásmu 868MHz

je potřeba anténu větší a také by bylo vhodné dosáhnout co nejlepších parametrů, a proto byl vyveden SMA konektor pro připojení externí antény.

4.3 Víko

Víko krabičky by mělo sloužit k zobrazení stavu zařízení pomocí LED diod a také je to jediné místo, které bude dostupné při osazení do rozvaděče a tudíž jediná možnost, kam umístit anténu pro pásmo 868MHz. Bylo by tedy vhodné, kdyby víko bylo průsvitné, aby bylo vidět LED diody, ale zároveň neprůhledné, aby nebylo vidět útroby zařízení. Vybraná krabička se standardně dodává bez horního víka. Je možné k ní dokoupit víko průhledné, šedé nebo červené. Ani jedno z nich tedy není vyhovující.

Další problém je s osazením konektoru SMA pro anténu. Jedná se o závit, na kterém jsou dvě šestihranné matice, přičemž vnitřní je pevně spojená a vnější demontovatelná. Je tedy nutné mít z obou stran víka prostor na matici, který ale v navržené konfiguraci PCB není.



Obrázek 4.3: Náhled namodelovaného víka

Tak vznikla myšlenka na 3D tisk víka, které by bylo duté a vnitřní matice by do něj zapadla, zatímco z vnější strany by bylo vystouplé. Použití bílého plastu se navíc dá docílit neprůhlednosti, ale dostatečné průsvitnosti. Bylo proto v programu Autodesk Inventor navrženo víko stejných rozměrů jako originální, do kterého byl přidán otvor pro SMA konektor a také vybrání kolem LED diod pro lepší průsvitnost, viz obrázek 4.3. Tisk byl proveden z bílého PLA plastu na tiskárně Prusa i3 MK3.

4.4 Osazování

Osazování bylo provedeno ručně s pomocí pájecí pasty a horkého vzduchu, některé součástky byly pájeny mikropájkou. Z hlediska náročnosti osazení byl největší komplikací modul s RF transceiverem – malé plošky jsou vyvedeny pouze zespodu, nikoliv z boku jako procesorový modul. Byl tedy zvolen postup, kdy se nejprve pocínovaly plošky jak na modulu, tak na PCB, poté se k sobě plošky přiložily a pro spojení bylo použito horkého vzduchu. U procesorového modulu byly nejdříve pomocí pájecí pasty zapájeny velké plošky pod modulem určené k odvodu tepla, následně byly kontakty po obvodu zapájeny mikropájkou.

4.5 Cena

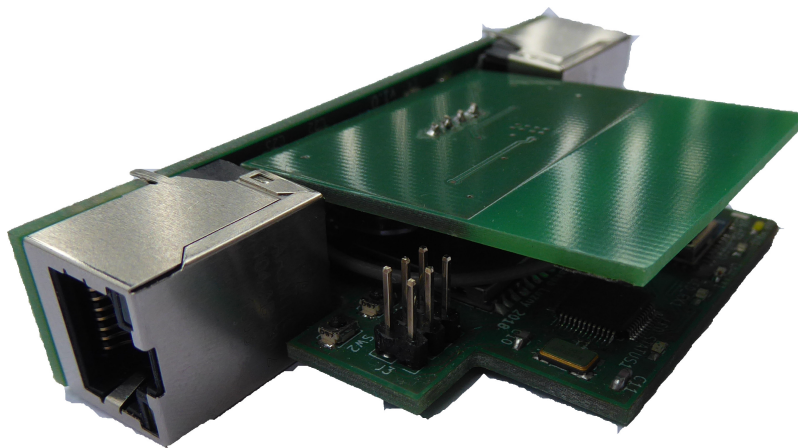
Celková cena kupovaných součástek na jeden kus prototypu dosáhla 1700Kč, viz. příloha D. Tato suma by se dala snížit nahrazením některých součástek levnějšími alternativami a nákupem ve větším množství. K této ceně je nutné přičíst náklady na výrobu plošných spojů, které činily v případě prototypu cca 800Kč na zařízení. Ty by však šly snížit v případě výroby více kusů velmi významně. Cena plastového filamentu použitého na 3D tisk víka se bude pohybovat v řádu jednotek korun.

V prototypu bylo osazení prováděno ručně, v případě sériové výroby by ale bylo vhodné jej provádět strojově. Cena strojového osazení by byla nutná individuálně poptat a to nebylo v rámci této práce provedeno.

4.6 Zjištěné nedostatky

Jak je u prvních prototypů běžné, byly po kompletaci zařízení zjištěny nějaké nedostatky. Prvním nedostatkem je, že po sestavení dohromady nedoléhá zdrojová PCB přímo na tu propojovací, ale na osazené kondenzátory. To způsobí, že zdrojová PCB přechází ven z krabičky. Tento problém byl vyřešen odbroušením asi 1mm zdrojové desky. V případné další revizi by bylo vhodné kondenzátory umístit jinak. V návrzích přiložených na paměťovém médiu byla PCB zmenšena.

Další nedostatek se nachází už v samotném schématu zapojení. Při výměně konektorů RJ45 za jiný typ v pozdější fázi vývoje byly opomenuty rezistory pro omezení proudu protékajícího signalizačními LED diodami. Na prototypu proto nebyly tyto diody vůbec zapájeny kvůli riziku poškození procesoru nadměrným odebíraným proudem. Ve schématu, které je součástí příloh této práce, je toto již opraveno.



Obrázek 4.4: Fotografie sestavených a osazených PCB.

Kapitola 5

Firmware

Od výrobce byl ve Flash paměti procesorového modulu nahrán bootloader U-Boot, který zaváděl upravenou linuxovou distribuci OpenWRT ve verzi 14.07. Tato verze již není podporována, a tudíž by bylo komplikované do ní doinstalovávat další software. Proto bylo rozhodnuto o aktualizaci na novější verzi. Bootloader byl ponechán beze změny.

5.1 OpenWRT

Jako základ firmwaru byla použita aktuální vývojová verze z větve *master* gitového repozitáře projektu. Poslední produkční verze byla totiž vydána začátkem roku 2017 ještě jako LEDE a přišlo mi vhodné začít vývoj nového projektu na aktuální verzi, která pravděpodobně v dohledné době dojde k produkčnímu vydání.

Pro správnou funkčnost zařízení bylo nutné do OpenWRT přidat jeho podporu. To ovšem díky podpoře platformy jako takové nezabralo mnoho času. Po přidání nového zařízení do Makefile projektu bylo nutné také vytvořit jeho Device Tree (DTS) soubor. Jedná se o standardní formát, v jakém se v posledních verzích Linuxu popisuje hardwarová konfigurace zařízení. Definuje tedy např. množství připojené paměti RAM, vyvedená rozhraní, zařízení připojená na sběrnících (jako je Flash paměť na SPI) a multiplexování pinů. Při jeho tvorbě jsem vycházel z DTS souboru pro platformu LinkIt Smart, jejíž podpora již v OpenWRT byla. Dále je třeba upravit startovací skript *network*, ve kterých se nastavuje zapojení Ethernetového switchu, tzn. které porty jsou na daném zařízení vyvedeny a které mají být mezi sebou propojeny. Zde bylo nastaveno, že oba porty jsou switchem propojené a tato síť je označena jako LAN. V dalším skriptu se nastavovalo, které LED diody mají sloužit k indikaci stavu zařízení. Modrá LED dioda byla nastavena jako stavový indikátor, zatímco bílá LED signalizuje aktivitu bezdrátového rozhraní WiFi.

5.2 Problém s Flash pamětí

Při vývoji se narazilo na problém s restartem zařízení. Po té, co na zařízení naběhl Linux, nešlo zařízení restartovat. Nezávisle na tom, zda byl restart vyvolán příkazem *reboot* z linuxové konzole, nebo hardwarově tlačítkem, zařízení se zaseklo. Jediný způsob, jak obnovit jeho funkčnost, bylo odpojit a znovu připojit napájení. Pokud byl ale procesor restartován ještě v bootloaderu, naběhl správně. Stejně tak ve verzi OpenWRT nahrané výrobcem modulu. Po chvíli bádání bylo objeveno diskuzní fórum[6], kde se řešilo podobné chování. To

nás přivedlo k myšlence, že je problém způsobený neshodou mezi délkou adresy zasílanou procesorem po restartu a tím, co Flash paměť očekává.

Standardně se totiž paměti typu NOR připojené pomocí rozhraní SPI adresují pomocí 3 bajtů adresy. Adresují se přímo jednotlivé uložené bajty, ve větších blocích probíhá pouze mazání paměti. To umožňuje adresovat celkem 2^{24} bajtů, což odpovídá 16MiB. Proto bylo pro větší paměti zavedeno adresování pomocí 4 bajtů, umožňující adresovat 2^{32} bajtů (4GiB). Pro zachování kompatibility ovšem i tyto větší paměti podporují starší metodu adresování. K přepínání mezi nimi existují dva přístupy. Stavový, kdy se paměť speciální příkazem přepne do režimu 4B adresování a všechny následující příkazy očekávají 4B adresu, dokud se jiným příkazem nepřepne zpět. A také bezstavový, kdy paměť kromě běžných příkazů pro čtení/zápis obsahuje další sadu těchto příkazů, které očekávají 4B adresu. V případě stavového přístupu se různé typy paměti liší tím, který adresní mód je výchozí po jejich zapnutí. Některé na to dokonce mají nevolatilní konfigurační registr.

Procesor MT7688 po startu začne číst kód z paměti Flash. K tomu potřebuje znát délku adresy, kterou má paměti posílat. Dle datasheetu tuto informaci procesor získá přečtením logické úrovně na pinu SPI_CS1. V případě vysoké úrovně pak začne posílat adresy dlouhé 4B, v případě nízké úrovně 3B. Na modulu HLK-7688A je osazen pull-down rezistor, který zajistí nízkou úroveň a procesor proto adresuje tříbajtově.

Konkrétní paměť použitá na modulu je typu MX25L25635F výrobce Macronix. Ta je po startu v režimu 3B adresování a podporuje jak stavový, tak bezstavový přístup ke 4B adresám. V Linuxu ovšem byla detekována jako typ MX25L25635E. To je dle Application Note výrobce[14] starší typ, který nepodporuje bezstavovou 4B adresaci. Proto ji linuxové jádro při startu přepne do režimu 4B adresace. Při restartu v tomto režimu paměť zůstane, ale procesor bude posílat adresy dlouhé 3B, takže dojde k jeho zacyklení.

Linuxové jádro používá pro zjištění typu paměti příkaz JDID. Na ten odpoví oba typy paměti stejnou odpovědí a tudíž není možné rozlišit typ MX25L25635F od staršího MX25L25635E. Proto je v ovladači definován starší typ bez podpory bezstavové 4B adresace a dochází k přepínání na 4B mód. Toto bylo vyřešeno patchem do tohoto ovladače, který změnil typ detekované paměti na MX25L25635F a nastaví příznak o podpoře bezstavové 4B adresace. Po tomto zásahu lze zařízení kdykoliv spolehlivě restartovat.

5.3 Ovladač WiFi

Pro funkčnost bezdrátové části čipu MT7688 v operačním systému Linux je nutný ovladač, který zajistí komunikaci jádra s hardwarem a propojení na vyšší síťové vrstvy. Pro tento čip existují dva ovladače. Jeden dodává přímo MediaTek a je uzavřený, nejsou k němu dostupné zdrojové kódy. Ten je dodáván ve výchozí instalaci jak v referenční platformě LinkIt Smart, tak i v modulu HLK-7688A. Je navrhnut tak, aby kromě Linuxu podporoval i jiné operační systémy a nepoužívá standardní linuxové API pro svou konfiguraci. Kvůli jeho uzavřenosti je problém s jeho fungováním na novějších verzích jádra, než pro které byl sestaven. To se snaží řešit projekt mtk-openwrt-feeds uživatele Nossiac na GitHubu[4]. Tento uživatel má přístup ke zdrojovým kódům oficiálního ovladače a vydává jeho sestavení s novějšími verzemi jádra. Projekt má formu tzv. package feedu pro OpenWRT. Jedná se o strom předpisů pro sestavení balíčků, který lze snadno přidat do sestavení OpenWRT. Kromě samotného ovladače obsahuje také další podpůrné nástroje, které převádí nastavení z UCI na formát, který ovladač zpracovává a samostatný modul do LuCI pro jeho konfiguraci.

Kvůli uzavřenosti a problémům s kompatibilitou začali vývojáři OpenWRT vyvíjet otevřený ovladač s názvem mt76[5], který je standardní součástí projektu OpenWRT. Ten je však stále ve fázi vývoje a není příliš doporučován pro produkční nasazení.

V tomto zařízení je použit ovladač mt76, který má ovšem problém s častým odpojováním, jak v režimu AP, tak klient. Byl také testován oficiální ovladač, ale s tím se podařilo pouze vyhledat síť, připojit už nikoliv. Předpokládá se, že při použití stabilní verze OpenWRT by oficiální ovladač výrobce fungoval spolehlivě.

5.4 MCU

Pro vývoj firmwaru pro mikrokontrolér bylo použito vývojové prostředí Atollic TrueSTUDIO® for STM32 ve verzi 9.0.0. Jedná se o výsledek spolupráce mezi firmami STMicroelectronics a Atollic a je možné jej zdarma používat pro programování mikrokontrolérů řady STM32 bez jakýchkoliv omezení. Toto prostředí je založené na Eclipse.

Pro usnadnění vývoje nabízí STMicroelectronics nástroj STM32CubeMX. Ten po výběru čipu vykreslí jeho pouzdro a umožní uživateli graficky nastavit funkce jednotlivých pinů a jejich pojmenování. Podobně je schopný zobrazit schéma hodinových signálů v mikrokontroléru a umožňuje uživateli nastavit jednotlivé děličky a multiplexy. Následně program umí vygenerovat kód v jazyce C, který obsahuje inicializace nadefinovaných periférií a pinů. K tomuto kódu umí také vytvořit projekt pro některá IDE (např. zmíněné TrueSTUDIO) nebo obyčejné Makefile.

Součástí vygenerovaného projektu je také knihovna pro hardwarovou abstrakci periférií procesoru. Lze vybrat ze dvou variant – knihovny HAL (Hardware Abstraction Layer), která poskytuje vyšší míru abstrakce díky udržování stavových proměnných a tím umožňuje také snadnější přenos programu na jinou řadu mikrokontroléru. Alternativou je knihovna LL (Low Level), která poskytuje pouze základní abstrakci a neudrží si vlastní stav – jedná se víceméně o sadu krátkých funkcí, které zapisují nebo čtou registry.

Původně byl program vyvíjen s pomocí knihovny HAL, avšak docházelo k náhodnému tuhnutí programu při komunikaci s RF transceiverem přes rozhraní SPI. Bylo zaujato podezření, že dochází k přepisu paměti někde uvnitř této knihovny. Vzhledem k její těžkopádnosti snaha o nalezení problémů selhala a vyzkoušelo se program přepsat s pomocí knihovny LL. K tomu bylo potřeba naprogramovat funkce pro abstrakci komunikace pomocí rozhraní SPI a UART. Poté už program běžel stabilně. Avšak i v knihovně LL se vyskytují chyby - např. funkce pro nastavení komunikační rychlosti rozhraní UART místo zadané rychlosti zapíše do registru nulu, takže je potřeba rychlost nastavit ručně zápisem do registru.

V rámci demonstrační aplikace byl implementován příjem protokolu Wireless M-BUS. K tomu bylo využito kódu dodaného vedoucím práce, který vychází z Application Note k čipu CC1101.[21] Ta ukazuje metodu, jak přijímat dlouhé pakety Wireless M-BUS, které mohou dosahovat délky přes 300 bajtů pomocí paketového režimu RF transceiveru, který podporuje pakety maximální délky 255 bajtů. Čip CC1101 má také pouze 64bajtové FIFO, takže je nutné využívat přerušování rádia signalizující zaplňující se FIFO a data vyčítat průběžně.

5.5 Demonstrační aplikace

Pro předvedení možností platformy bylo třeba vytvořit demonstrační aplikaci. Byla proto implementována aplikace, která umožňuje pomocí webového rozhraní nahrát program do

mikrokontroléru a také webové rozhraní, které bude zobrazovat dekodované základní údaje z přijatých paketů Wireless M-Bus.

5.5.1 Výběr programovacího jazyka

Díky použití distribuce OpenWRT je možné si vybrat ze široké škály jazyků, ve kterých se dá zařízení programovat. Základními jazyky linuxového světa jsou bez sporu jazyky C a C++. Tyto jazyky se však nejeví příliš vhodné na vytvoření demonstrační webové aplikace. Bylo by vhodnější použít spíše nějaký interpretovaný jazyk, který umožní úpravy aplikace přímo na zařízení, ale také spuštění aplikace na počítači bez nutnosti překompilování. Odpadá tím také nutnost mít pro vývoj aplikace instalované OpenWRT SDK.

Zajímavé by bylo napsat aplikaci v jazyce Lua a integrovat s webovým rozhraním LuCI. Toto však bylo zavrženo z důvodu nulové zkušenosti s jazykem a také by to znamenalo obtížné spuštění aplikace mimo zařízení.

Z dalších jazyků se jevílo zajímavé napsat aplikaci pomocí NodeJS. Jedná se o implementaci JavaScriptu, která je postavená na enginu V8 z webového prohlížeče Google Chrome. Hojně využívá model událostí a asynchronní I/O operace, díky čemuž minimalizuje režii a maximalizuje výkon. Disponuje také balíčkovacím systémem *npm*, který je největším ekosystémem open-source aplikací na světě.

S NodeJS se ovšem narazilo na dva problémy. Tím prvním je, že se nepodařilo sestavit knihovnu *node-serial* pro komunikaci přes rozhraní UART. Tím druhým je prakticky neexistence standardní knihovny. Ta se projevuje tak, že každý nainstalovaná knihovna z *npm* si s sebou nese veškeré svoje závislosti. Často si tak kvůli jedné několikařádkové metodě přinese jako závislost celou knihovnu. To je ještě umocněno tím, že *npm* standardně ke každé knihovně instaluje kromě zdrojových kódů také dokumentaci, testy apod. Ve finále pak například webový framework ExpressJS zabere 2MiB na úložišti, což je v tomto zařízení poněkud omezující. Tento problém je zmíněn i v knize *Node.js for Embedded Systems*.^[18] Dal by se řešit připojením další paměti přes rozhraní SPI, použití SD karty není v režimu procesoru *IoT Gateway* možné. Došlo by tak ovšem ke zvýšení ceny zařízení, v případě SD karty je zde i riziko snížené spolehlivosti.

Proto bylo rozhodnuto o použití jazyka Python. Ten si s sebou nese rozsáhlou standardní knihovnu, díky čemuž není nutné k aplikaci doinstalovávat tolik závislostí. Součástí OpenWRT jsou jak balíčky Pythonu 2, tak novějšího Pythonu 3. Vzhledem k tomu, že se jedná o nový projekt byla použita novější verze 3. Pro tu ale nebyl v OpenWRT balíček knihovny *pyserial* pro komunikaci přes rozhraní UART, takže bylo nutné jej vytvořit.

5.5.2 Implementace

Demonstrační aplikace má formu webové stránky a tvoří ji dvě součásti – tzv. *frontend* a *backend*. Backend je ta část aplikace, která běží na zařízení a komunikuje s mikrokontrolérem. Data získaná z mikrokontroléru poté zpracovává, z hlaviček datových paketů extrahuje užitečná data. K tomu byla implementována vlastní knihovna pro dekodování hlaviček protokolu M-BUS, která vychází z dostupné specifikace^[17]. Po dekodování jsou data předána webové aplikaci, která je odešle ve formátu JSON. Pro implementaci webové aplikace byl použit framework Bottle¹. Jedná se o minimalistický webový framework, celý je implementován v jednom zdrojovém souboru. Ten není distribuovaný s aplikací, používá se balíček

¹ <https://bottlepy.org>

systému OpenWRT. Součástí backendu je také komunikace s bootloaderem mikrokontroléru STM32. K tomu byla využita knihovna *stm32bl*², která je distribuovaná s aplikací.

Frontend naopak běží ve webovém prohlížeči a skládá se ze samotné šablony v jazyce HTML, kaskádových stylů CSS a skriptů v jazyce JavaScript. Na webové stránce se hojně využívá komponentů CSS frameworku Bootstrap³. Skript pro získávání dat z backendu a jejich vykreslování využívá prezentačního frameworku Vue.js⁴.

Aplikace se spouští automaticky po startu zařízení a poslouchá na portu 8080. Standardní HTTP port 80 je totiž vyhrazen konfiguračnímu rozhraní OpenWRT LuCI. Snímek obrazovky aplikace po přijetí několika paketů je v příloze B.

5.5.3 Komunikace s MCU

Komunikační protokol je jednosměrný, textový a jednotlivé zprávy jsou zakončený znakem nového řádku. Po sériové lince se data přenáší rychlostí 230 400 baudů za sekundu, která se ukázala být dostatečná pro potřeby aplikace. Jde však v případě potřeby zvýšit, testována byla rychlost 1MBaud/s.

První znak každé zprávy určuje její typ. V případě, že začíná znakem #, jedná se o ladící výpis. Pakety úspěšně přijaté na sběrnici WM-Bus začínají písmenem podle typu zprávy, tzn. T nebo C. Následuje obsah paketu, zakódovaný jako řetězec jednotlivých bajtů převedený do šestnáctkové soustavy (tzv. hexdump). To jsou také jediná data, která aplikace zpracovává. V případě, že po dekodování paketu nevyšel jeho kontrolní součet, jedná se o zprávy t nebo c. Pokud se data ani nepodařilo dekodovat, jedná se o paket RAW, označení r.

5.6 Nahrání firmwaru do MT7688

Nahrát firmware pro hlavní procesor MT7688 lze třemi způsoby. Nejjednodušší metoda je z běžícího systému přes webové rozhraní nebo konzoli. Pokud systém nenaběhne, je možné využít bootloader, který je v modulu nahrán od výrobce a umí stáhnout obraz firmwaru ze sítě pomocí protokolu TFTP. V případě, že by došlo k poškození bootloaderu, lze firmware nahrát pomocí externího zařízení, které je schopné komunikovat po SPI. Toto bude předvedeno na minipočítači Raspberry Pi.

5.6.1 Z běžícího systému

Součástí systému OpenWRT je webového rozhraní LuCI, které poslouchá na standardním HTTP portu 80. Po přihlášení je možné v sekci System → Backup/Flash firmware nahrát obraz firmwaru.

Alternativou je připojení k zařízení protokolem SSH a aktualizace z příkazové řádky. K tomu je nutné soubor nejprve do zařízení nahrát, použít lze např. protokol SFTP. Firmware lze poté zapsat příkazem:

```
sysupgrade <název souboru>
```

² <https://github.com/pavelrevak/stm32bl>

³ <https://getbootstrap.com/>

⁴ <https://vuejs.org/>

5.6.2 S pomocí bootloADERu

Pro nahrání firmware přes bootloADER je třeba počítač schopný komunikace po sériové lince na logických úrovních 0-3,3V. Běžný počítač se dá doplnit převodníkem z USB. Dále je nutné mít v lokální síti TFTP server, na který se uloží obraz firmware. Lze také použít jakýkoliv počítač, na kterém se spustí implementace TFTP serveru, spousta jich je volně k dispozici.

Po nahrání obrazu na TFTP server, připojení zařízení do sítě (libovolným konektorem) je nutné propojit signály RX, TX a GND, které jsou dostupné na konektoru J1 zařízení, s počítačem (nebo USB převodníkem). Datové signály je nutné při propojení překřížit, tzn. signál TX zařízení jde na signál RX počítače a obdobně pro druhý signál. Následně se na počítači spustí aplikace pro komunikaci po sériové lince (v GNU/Linuxu například *picocom*) a nastaví parametry spojení:

- komunikační rychlost 57600 baud
- 8 datových bitů
- bez parity
- 1 stop bit

Po restartu zařízení (stisknutím tlačítka SW1) by se měl v terminálu objevit výpis bootloADERu. Je nutné během 5 sekund stisknout volbu 2 „Load system code then write to Flash via TFTP“, poté potvrdit přepsání paměti volbou Y a následně zadat IP adresu zařízení, TFTP serveru a název souboru s firmwarem. Následně se spustí stahování souboru, průběh je indikován výpisem symbolu # na terminál. Po jeho dokončení se spustí samotný zápis do paměti, kde je průběh indikován výpisem teček. Dokončení je oznámeno větou „Done.“ a následuje spuštění nahraného systému.

5.6.3 S pomocí Raspberry Pi

V případě, že dojde k přepsání bootloADERu, lze využít sběrnice SPI vyvedené na konektoru J1 pro přístup k Flash paměti z externího zařízení. Takovým zařízením může být např. minipočítač Raspberry Pi v kterékoliv variantě, který bude použit v následujícím postupu.

Nejprve je nutné propojení dle tabulky 5.1. Poté je potřeba povolit použití SPI na Raspberry. Toho lze docílit odkomentováním řádku `dtoverlay=spi=on` v souboru `/boot/config.txt` a následným restartem. Pro práci s pamětí Flash lze použít program *flashrom*. Ten je sice standardní součástí balíčkovacího systému distribuce Raspbian, ale v době psaní práce byl použitý Flash čip podporován pouze vývojovou verzí. Ta se nachází v gitovém repozitáři, stejně jako soubor README⁵, který popisuje postup pro instalaci.

Při jakékoliv práci s pamětí je třeba držet procesor v resetu. Toho lze docílit buď držením tlačítka SW1, což je poněkud nepohodlné, a nebo propojením pinu 3 konektoru J1 s GND. Přechzení paměti a zapsání obsahu do souboru lze provést následujícím příkazem:

```
flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=1000 -r <název souboru>
```

Naopak zapsání souboru do paměti zařízení příkazem:

```
flashrom -p linux_spi:dev=/dev/spidev0.0,spispeed=1000 -w <název souboru>
```

⁵ <https://review.coreboot.org/cgit/flashrom.git/tree/README>

Číslo pinu GPIO Raspberry	Číslo pinu J1	Název signálu
19	4	MOSI
21	2	MISO
23	5	SCK
24	1	CS
25	6	GND

Tabulka 5.1: Zapojení signálů pro programování Flash paměti z Raspberry Pi

Číslo pinu ST-Link	Číslo pinu J3	Název signálu
2	1	SWDIO
4	4	GND
6	3	SWCLK

Tabulka 5.2: Zapojení signálů pro programování mikrokontroléru pomocí ST-Link

Soubor obsahuje kompletní obsah paměti, tzn. začíná bootloaderem a pokračuje systémem OpenWRT a uživatelskými daty. Postup jeho vytvoření není součástí této práce. Lze jej však získat vyčtením z jiného funkčního zařízení.

5.7 Nahrání firmwaru do MCU

Firmware do mikrokontroléru lze nahrát dvěma způsoby. Standardně se o to stará uživatelská aplikace běžící pod systémem OpenWRT, která k tomu využívá bootloader integrovaný v mikrokontroléru a dostupný na rozhraní USART1. V případě, že žádná taková aplikace neběží, lze firmware nahrát z příkazové řádky systému OpenWRT. Následující příkaz zajistí nahrání, ověření a spuštění programu:

```
stm32flash -w <soubor s programem> -i -11,11 -v -g 0x0 /dev/ttyS1
```

Během vývoje je však výhodou mít možnost program krokovat a také snadno nahrát přímo z vývojového prostředí. Proto je na konektoru J3 vyvedeno rozhraní SWD. K tomu lze připojit tzv. debugger (např. ST-Link), který zajistí komunikaci mezi vývojovým prostředím a mikrokontrolérem. Popis zapojení debuggeru ST-Link se nachází v tabulce [5.2](#).

Kapitola 6

Závěr

V této práci se mi podařilo prozkoumat možnosti platformy MediaTek MT7688A, do detailu projít možnosti systémových modulů s tímto čipem a jejich softwarovou podporu. Na základě těchto poznatků jsem navrhl a sestavil funkční prototyp vestavěného zařízení, které může sloužit jako platforma pro nejrůznější aplikace internetu věcí. Je schopné komunikace v pásmu ISM, zpracování dat a jejich odeslání pomocí počítačové sítě. Zařízení je levné na výrobu, kompaktní a vykazuje nízkou spotřebu elektrické energie. Pro toto zařízení jsem sestavil operační systém OpenWRT, díky čemuž se zařízení stalo snadno softwarově rozšiřitelné.

Pro demonstraci této platformy jsem vytvořil program pro mikrokontrolér, který za pomoci připojeného RF transceiveru přijímá data pomocí bezdrátového standardu Wireless M-BUS, dekóduje datové rámce a ty posílá ke zpracování do demonstrační aplikace. Ta z hlaviček těchto paketů dekóduje identifikační data vysílajícího zařízení a vykreslí je na webové stránce.

Vytvořené zařízení skrývá rozsáhlé možnosti dalšího rozvoje, zejména po softwarové stránce. Je na něm možné implementovat aplikaci, která by byla schopná přijímat data vysílaná různými protokoly a tyto data centralizovat do jednoho bodu pro další zpracování. Zajímavou alternativou by také mohla být portace nějakého existujícího softwarového řešení na vytvořené zařízení.

Literatura

- [1] *1M XTS Modulbox XTS / Italtronic*. [Online; navštíveno 30.04.2018].
URL http://eng.italtronic.com/products/modulbox_xts_en/1m_xts_modulbox_xts_en/?prodotto=25.0104RJ0.BL
- [2] *About the OpenWrt/LEDE project*. [Online; navštíveno 29.04.2018].
URL <https://openwrt.org/about>
- [3] *Blue Pill - STM32duino wiki*. [Online; navštíveno 30.04.2018].
URL http://wiki.stm32duino.com/index.php?title=Blue_Pill
- [4] *Nossiac/mtk-openwrt-feeds: mtk-openwrt-feeds – GitHub*. [Online; navštíveno 09.05.2018].
URL <https://github.com/Nossiac/mtk-openwrt-feeds>
- [5] *openwrt/mt76: mac80211 driver for MediaTek MT76x2e, MT7603, MT7628 and MT7688 – GitHub*. [Online; navštíveno 09.05.2018].
URL <https://github.com/openwrt/mt76>
- [6] *Reset AI7688H in bootloader - LinkIt Smart 7688 - MediaTek Labs Forums*. [Online; navštíveno 09.05.2018].
URL <https://en.forum.labs.mediatek.com/t/reset-ai7688h-in-bootloader/251/20>
- [7] *ReSpeaker Core*. [Online; navštíveno 29.04.2018].
URL http://wiki.seeedstudio.com/ReSpeaker_Core/
- [8] *U-Boot README*. [Online; navštíveno 08.05.2018].
URL http://git.denx.de/?p=u-boot.git;a=blob_plain;f=README;hb=HEAD
- [9] *MediaTek MT7688 Datasheet*. Duben 2016, [Online; navštíveno 30.04.2018].
URL <https://labs.mediatek.com/en/download/50WkbgbH>
- [10] *Abracon: RJ45 SINGLE PORT 100/1000/2.5G/5G BASE-T MAGNETICS*. [Online; navštíveno 01.05.2018].
URL <https://abracon.com/Magnetics/ARJM11.pdf>
- [11] *AcSip: AI7688H*. [Online; navštíveno 01.05.2018].
URL <http://www.acsip.com.tw/index.php?action=products-detail&fid1=11&fid2=&fid3=&id=29>
- [12] *Anaren: Anaren Integrated Radio – A1101R08x User's Manual*. [Online; navštíveno 30.04.2018].

- URL https://cdn.anaren.com/product-documents/AIR/ProprietaryRF/A1101R08X/A1101R08x_UserManual.pdf
- [13] Hi-LINK: *wifi module HLK-7688A support OpenWRT with 32Mflash 128M RAM ,free SDK*. [Online; navštíveno 01.05.2018].
URL http://www.hlktech.net/product_detail.php?ProId=69
- [14] Macronix: *Application Note: Migrating from MX25L25635E/25735E to MX25L25635F/25735F*. [Online; navštíveno 09.05.2018].
URL http://www.macronix.com/Lists/ApplicationNote/Attachments/1892/AN0200V1_MGRT_MX25L25635E_25735E%20to%20MX25L25635F_25735F.pdf
- [15] MediaTek: *LinkIt Smart 7688 Resources/FAQ*. [Online; navštíveno 30.04.2018].
URL <https://docs.labs.mediatek.com/resource/linkit-smart-7688/en/faq>
- [16] MediaTek: *LinkItTM Smart 7688*. [Online; navštíveno 30.04.2018].
URL <https://labs.mediatek.com/en/platform/linkit-smart-7688>
- [17] Miehlisch, F.: *The M-Bus: A Documentation Rev. 4.8*. [Online; navštíveno 15.05.2018].
URL <http://www.m-bus.com/mbusdoc/default.php>
- [18] Mulder, P.; Breseman, K.: *Node.js for Embedded Systems: Using Web Technologies to Build Connected Devices*. O'Reilly Media, 2016, ISBN 978-1-491-92899-8.
- [19] Onion Corporation: *OMEGA2S Surface Mount Packaged version of the Omega2 for High Volume Commercial and Industrial OEMs*. [Online; navštíveno 30.04.2018].
URL <https://onion.io/omega2s/>
- [20] Onion Corporation: *Onion Omega2 Documentation*. [Online; navštíveno 30.04.2018].
URL <https://docs.onion.io/omega2-docs/omega2.html>
- [21] Seem, P.: *Application Note AN067: Wireless MBUS Implementation with CC1101 and MSP430*. [Online; navštíveno 09.05.2018].
URL <http://www.ti.com/lit/an/swra234a/swra234a.pdf>
- [22] STMicroelectronics: *AN2586 Application note: Getting started with STM32F10xxx hardware development*. [Online; navštíveno 30.04.2018].
URL http://www.st.com/resource/en/application_note/cd00164185.pdf
- [23] STMicroelectronics: *Medium-density performance line ARM®-based 32-bit MCU with 64 or 128 KB Flash, USB, CAN, 7 timers, 2 ADCs, 9 com. interfaces*. Srpen 2015, [Online; navštíveno 30.04.2018].
URL <http://www.st.com/resource/en/datasheet/cd00161566.pdf>
- [24] Texas Instruments: *CC1101 Low-Power Sub-1 GHz RF Transceiver*. Září 2013, [Online; navštíveno 30.04.2018].
URL <http://www.ti.com/lit/ds/symlink/cc1101.pdf>
- [25] Texas Instruments: *LMR16020 SIMPLE SWITCHER® 60 V, 2A Step-Down Converter*. Prosinec 2015, [Online; navštíveno 29.04.2018].
URL www.ti.com/lit/ds/symlink/lmr16020.pdf

Příloha A

Obsah přiloženého paměťového média

- demoapp/ – zdrojové kódy demonstrační aplikace
- mcu/ – zdrojové kódy a sestavený firmware pro mikrokontrolér
- openwrt/ – kompletní strom použitého OpenWRT a popis změněných souborů
- hw/ – schéma zapojení a návrh PCB
- viko/ – návrh víka krabičky
- latex/ – zdrojové soubory tohoto dokumentu
- xluzny00.pdf – elektronická verze tohoto dokumentu

Příloha B

Screenshot demonstrační aplikace

Wireless M-BUS Receiver x

172.16.1.2:8080

MT7688-IoT Choose firmware Browse Program MCU Reset MCU

Received Wireless M-BUS packets

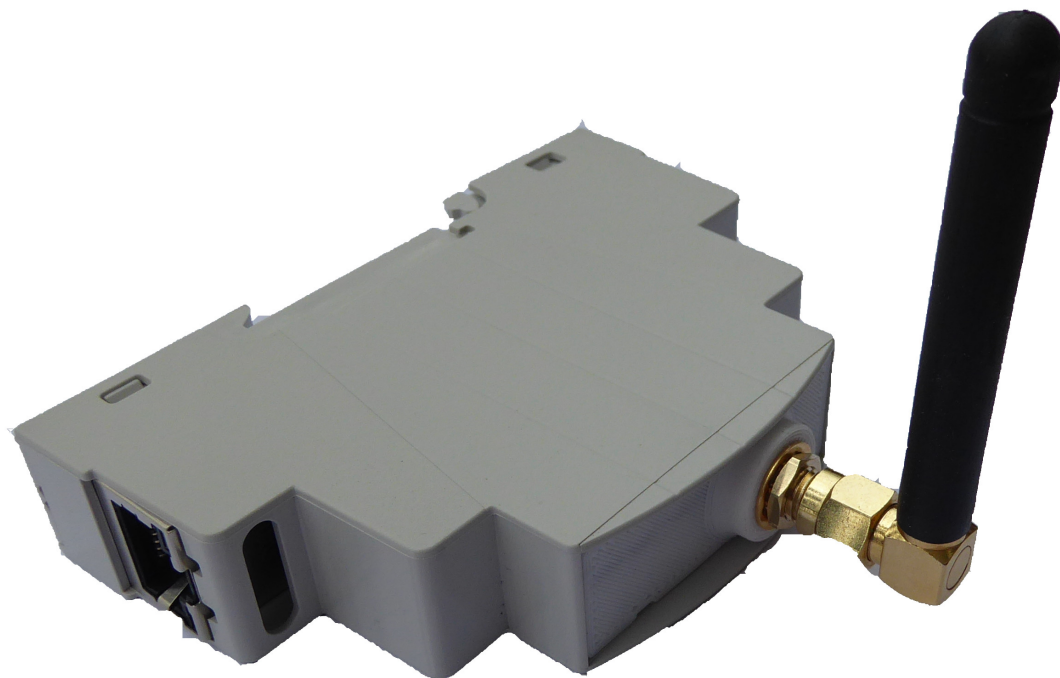
#	Manufacturer	Device ID	Version	Type (Medium)	Signal level
23	TCH	12307275	0x94	Unknown (0x80)	-87.5 dBm
22	TCH	12307167	0x94	Unknown (0x80)	-91.5 dBm
21	TCH	12307256	0x94	Unknown (0x80)	-92 dBm
20	TCH	12307254	0x94	Unknown (0x80)	-92.5 dBm
19	TCH	12307649	0x94	Unknown (0x80)	-88.5 dBm
18	APA	00882740	0x5	Water (0x7)	-91.5 dBm
17	TCH	12307282	0x94	Unknown (0x80)	-89 dBm
16	APA	00820977	0x5	Water (0x7)	-91 dBm
15	APA	00409472	0x5	Water (0x7)	-87 dBm
14	TCH	12307648	0x94	Unknown (0x80)	-86.5 dBm
13	APA	00670573	0x5	Water (0x7)	-91.5 dBm
12	TCH	12307275	0x94	Unknown (0x80)	-87.5 dBm
11	APA	00882673	0x5	Water (0x7)	-93.5 dBm
10	TCH	12307278	0x94	Unknown (0x80)	-88 dBm
9	APA	00820977	0x5	Water (0x7)	-88 dBm
8	TCH	12307656	0x94	Unknown (0x80)	-82 dBm
7	APA	00670573	0x5	Water (0x7)	-91.5 dBm
6	TCH	12307651	0x94	Unknown (0x80)	-88.5 dBm
5	APA	00409472	0x5	Water (0x7)	-88.5 dBm
4	TCH	12286681	0x94	Unknown (0x80)	-85.5 dBm
3	TCH	12307254	0x94	Unknown (0x80)	-91.5 dBm
2	TCH	12307649	0x94	Unknown (0x80)	-89.5 dBm
1	TCH	12307170	0x94	Unknown (0x80)	-97 dBm
0	TCH	12307281	0x94	Unknown (0x80)	-86 dBm

© Jakub Lužný 2018

Obrázek B.1: Screenshot demonstrační aplikace

Příloha C

Fotografie výsledného prototypu



Obrázek C.1: Fotografie výsledného prototypu

Příloha D

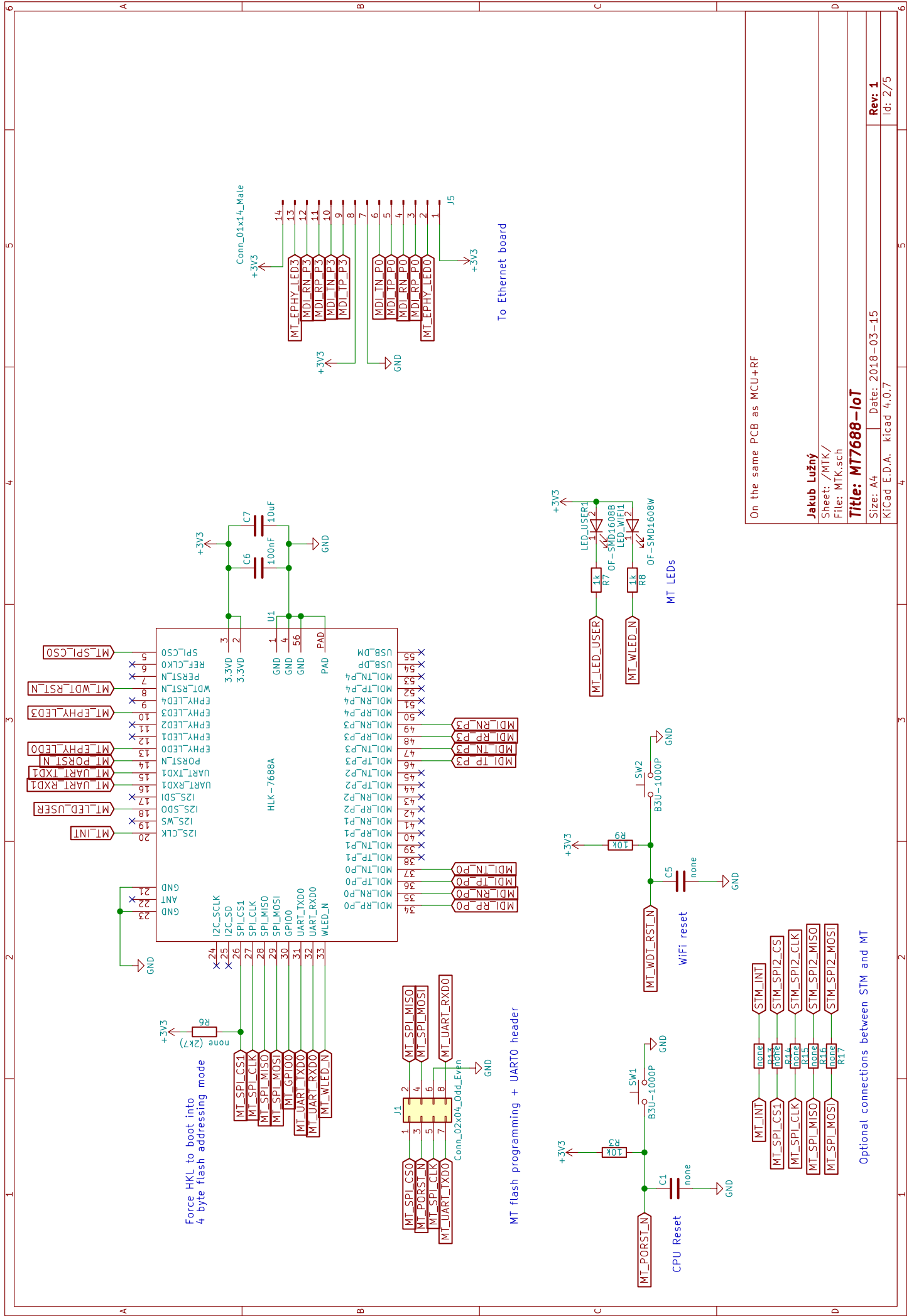
Seznam použitých součástek

Uvedená cena vždy reflektuje nejnižší dostačující množství limit. Všechny uvedené ceny jsou bez DPH. V celkové ceně není zahrnuta výroby plošných spojů, 3D tisku víka ani osazení.

Počet	Označení	Value	Dodavatel	Cena za kus [Kč]	Cena celkem [Kč]
2	C1, C5	neosazeno (0603)			0
15	C2, C3, C4, C6, C9, C12, C13, C16, C17, C18, C19, C21, C22, C23, C24	100nF (0603)	TME	0,2	3
1	C7	10uF (0603)	TME	2,6	2,6
1	C8	10uF/50V (1210)	TME	20	20
2	C10, C11	20pF (0603)	TME	0,2	0,4
2	C14, C20	1uF (0603)	TME	0,7	1,4
1	C15	150uF/10V (D)	TME	12,8	12,8
1	D1	S80	TME	4,9	4,9
1	D3	P6SMBJ28A (SMB)	TME	3,4	3,4
1	D4	SK23 (SMB)	TME	1,7	1,7
1	J1, J3	ZL202-80G	TME	11,2	11,2
1	J2	ZL305-04	TME	7	7
1	J4, J5	ZL211-50KG	TME	26,3	26,3
1	L1	SRN8040-330M (33uH)	Farnell	12,8	12,8
1	LED_RX1	OF-SMD1608G (0603)	TME	2,4	2,4
1	LED_STATUS1	OF-SMD1608RL (0603)	TME	2,4	2,4
1	LED_TX1	OF-SMD1608Y (0603)	TME	2,7	2,7
1	LED_USER1	OF-SMD1608B (0603)	TME	2,4	2,4
1	LED_WIFI1	OF-SMD1608W (0603)	TME	2,7	2,7
8	R1, R2, R4, R5, R27, R28, R29, R30	49,9 (0603)	TME	0,2	1,6
2	R3, R9	10k (0603)	TME	0,2	0,4
8	R7, R8, R11, R12, R23, R24, R25, R26	1k (0603)	TME	0,2	1,6
1	R10	47k (0603)	TME	0,2	0,2
7	R6, R13, R14, R15, R16, R17, R19	neosazeno (0603)			0
1	R18	0 (0603)	TME	0,2	0,2
1	R20	51k/1% (0603)	TME	0,2	0,2
1	R21	15k/1% (0603)	TME	0,2	0,2
2	SW1, SW2	B3U-1000P	Farnell	16,6	33,2
1	U1	HLK-7688A	Aliexpress	200	200
1	U2	LMR16020	Farnell	121,7	121,7
1	U3	STM32F103C8T6	TME	93,6	93,6
1	U4	A1101R08C	Farnell	456,1	456,1
2	U5, U6	ARJM11D7-104-AB-EW2	Digi-Key	139,1	278,2
1	Y1	ABM3B-8.000MHZ-10-1-U-T	TME	16,8	16,8
1	krabička	ITAL TRONIC 25.0104RJ0.BL	TME	58,1	58,1
1	pigtail kabel u.FL → SMA-F	Taoglas Limited CAB.719	Digi-Key	125,3	125,3
1	Anténa WiFi	AW-A2458G-FPC71U	SECTRON	125	125
1	Anténa 868MHz	GSM-ANT015	TME	67,5	67,5
				Celková cena	1700

Příloha E

Schéma zapojení



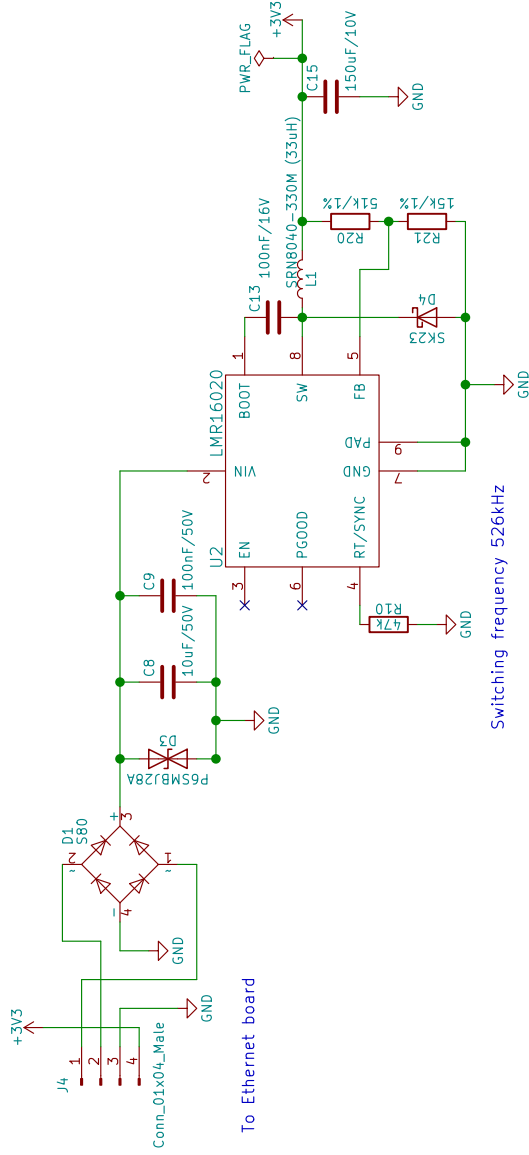
On the same PCB as MCU+RF

Jakub Lužný
Sheet: /MTK/
File: MTK.sch

Title: MT7688-IoT

Size: A4
Date: 2018-03-15
KiCad E.D.A. kicad 4.0.7

Optional connections between STM and MT



Jakub Lužný

Sheet: /Power/
File: Powersch

Title: MT7688-IoT

Size: A4 Date: 2018-03-15

KiCad E.D.A. kicad 4.0.7

Rev: 1

Id: 5/5